CLAIMS

We claim:

1. A method of generating an encrypted code in base L, comprising steps providing an n-bit raw number;

applying a one-way hash function on the n-bit raw number with a first secret key to generate a first strung;

designating an m-bit portion of the first string as an m-bit validation number; and

combining the m-bit validation number and the n-bit raw number to generate a second string.

2. The method of claim 1, further comprisingthe steps of:

aaplying a DES3 encryption algorithm to the second string with a second secret key to generate a third string; and

converting the third string to base L to generate the encrypted code.

3. The method of claim 1, wherein n=32, m=16, and L=29.

4. The method of claim 1, wherein the one-way hash function is MD5.

5. The method of claim 1, wherein the step of combining includes concatenating the m-bit validation number and the n-bit raw number.

6. The method of claim 1, wherein the m-bit validation number is the m most significant bit (MSB) portion of the second string.

7. The method of claim 1, wherein the m-bit validation number is the m most significant bit (MSB) of the first string.

8. A method of verifying the valifdity of a code, comprising the steps of:

generating a code with encrypted information;

fixing the code on an object to be distributed to a user;

obtaining the code from the object, by the user;

receiving the code on line from the user;

verifying the validity of the code by processing the encrypted information.

9. The method of claim 8, wherein the step of generating includes the steps of:

providing an n-bit raw number;

applying a one-way hash function on the n-bit raw number with a first secret key to generate a first string;

designating a m-bit portion of the first string as an m-bit validation number;

combining the m-bit validation number and the n-bit raw number to generate a second string;

applying a DES3 encryption algorithm to the second string with a second secet key to generate a third string; and

converting the third string to a base L to generate the code with the encrypted information.

10. The method of claim 9, wherein the step of verifying includes the steps of:

converting the code in base L to generate a first test code in base 2;

decrypting the first test code with the second secret key using a reverse DES3 encryption algorithm to generate a second test code;

applying the one-way hash algorithm to the second test code to generate a third test code; and

comparing a designated m-bit portion of the second test code to a designated m-bit portion of the third test code, and if the comparison is positive, declaring the code to be valid.

11. The method of claim 10, wherein the m-bit validation number is the m-most significant bit (MSB) of the first string in the generating step and the designated m-bit portion is the most significant bit portion of the second test code in the comparing step.

12. A method for awarding incentive points to a user, comprising the steps of:

generating a code with encrypted information;

providing the code to an entity for printing on an object;

receiving the code from a user on line, the code having been reterived from the object by the user;

verifying the validity of the code by processing the encrypted information.

13. The method of claim 12, wherein the step of generating includes the steps of:

providing an n-bit raw number;

applying a one-way hash function on the en-bit raw number with a first secret key to generate a first string;

designating an m-bit portion of the first string as an m-bit validation number;

combining the m-bit validation number and the n-bit raw number to generate a scond string;

applying a DES3 encryption algorithm to the second string with a second secret key to generate a third string; and

converting the third string to base L to generate ateh code with the encrypted information.

14. The method of claim 13, wherein the step of verifying includes:

converting the code in base L to generate a first test code in base 2;

decrypting the first test code with the second secret key using a reverse DES3 encryption algorithm to generate a second test code;

applying the one-way hash algorithm to the second test code to generate a third test code; and

comparing a designated m-bit portion of the second test code to a designated m-bit portion of the third test code, and if the comparison is positive, declaring the code to be valid.

15. The method of claim 14, wherein the step of verifying includes the steps of:

converting the code in base L to generate a first test code in base 2;

decrypting the first test code with the second secret key using reverse DES3 encryption algorithm to generate a second test code;

applying the one-way hash algorithm to the second test code to generate a third test code; and

comparing a designated m-bit portion of the second test code to a designated m-bit portion of the third test code, and if the comparison is positive, declaring the code to be valid.

16. The method of claim 15, wherein the m-bit validation number is the m most significant bit (MSB) of the first string in the generating step and the designated m-bit portion is the most significant bit portion of the second test code and third test code in the comparing step.

17. An offline-online points system, comprising:

a main server configured for providing a user with an interface to submit a code, wherein the code is obtainable offline and is associated with N points, wherein each point, characterized as a purchase or attention incentive point, is redeemable and maintainable in an account for the user;

a code server configured for maintaining valid codes and verifying, against the valid codes, that the code submitted by the user is valid such that a balance in the account for the user is increased by a predetermined number of points if the code is valid;

means for generating the code; and

means for fixing the code onto a medium such that the code is obtainable from the medium offline.

18. The offline-online points system of claim 17, wherein the means for generating the code includes

means for providing a number portion,

means for deriving a validation portion from the number portion,

means for appending the validation portion to the number portion to form a string,

means for encrypting the string, and

means for deriving the code from the encrypted string by converting the encrypted string to base L string.

19. The offline-online points system of claim 18, wherein the code is a fixed-length string with C characters, and wherein the means for deriving the code further includes means for prepending a character to the base L string any number of times that is needed to achieve the fixed-length of C characters.

20. The offline-online points system of claim 18, wherein L is the number of characters in the alphabet.

21. The offline-online points system of claim 18, wherein the string is 48-bits long and the number portion is 32-bits long.

22. The offline-online points system of claim 17, wherein the means for generating the code includes

means for providing a number portion, $S1_{INT}$, from a first string, S1

means for arranging a first secret key, K1, next to the number portion, $S1_{INT}$, from S1, to form a second string, S2,

means for applying a hash function to S2 to produce a third string, S3,

means for extracting a validation portion, $S1_{VAL}$, from S3 and arranging $S1_{VAL}$, next to $S1_{INT}$ in S1 ($S1 = S1_{VAL} + S1_{INT}$),

means for encrypting S1 using a second secret key, K2, to form a fourth string, S4, and

means for deriving the code by converting S4 to a base L fixed-length code string.

23. The offline-online points system of claim 22, wherein the first and second secret keys, K1 and K2, are 128-bits long and the encryption means includes DES3 encryption algorithm.

24. The offline-online points system of claim 22, wherein the hash function application means has MD5, a one-way hash algorithm.

25. The offline-online points system of claim 22, wherein S1 is 48-bits long and the number portion, $S1_{INT}$, is 32-bits long.

26. The offline-online points system of claim 17, wherein for verifying the submitted code the code server includes,

means for converting the submitted code from a base L string into a base 2 string, $S4_{BASE2}$,

means for decrypting $S4_{BASE2}$ using a second secret key, K2, to form a decrypted first string, S1',

means for providing a number portion, $S1'_{INT}$, from S1'

means for arranging a first secret key, K1, next to the number portion, $S1'_{INT}$, from S1, to form a second string, S2',

means for applying a hash function to S2' to form a third string S3',

means for extracting a validation portion from S3' and a validation portion from S1', and

means for determining if the code is valid by comparing the validation portion from S3' with the validation portion from S1'.

27. The offline-online points system of claim 26, wherein S3' and S1 are each 48-bits long and the secret keys, K1 and K2 are 128-bits long.

28. The offline-online points system of claim 26, wherein the decryption means includes $DES3^{-1}$ decryption algorithm and the hash function application means includes MD5 hash algorithm.

29. A method for offline-online handling of incentive points, comprising:

generating a code, wherein wherein the code is generated by providing a number portion, deriving a validation portion from the number portion, appending the validation portion to the number portion to form a string, encrypting the string, and deriving the code from the encrypted string by converting the encrypted string to base L string; and

fixing the code onto a medium such that the code is obtainable from the medium offline.

30. The method of claim 29, further comprising:

obtaining the code offline;

submitting the code online to a server that has valid codes, wherein the code is associated with N points maintained by the server in a user account, wherein each point, characterized as a purchase or attention incentive point, is redeemable; and

verifying the submitted code against the valid codes to determine if it is valid, wherein if the submitted code is valid, a predetermined number of points are added to the user account.

31. A method as in claim 29, wherein the code is a fixed-length string with C characters, and wherein a character is prepended to the base L string any number of times that is needed to achieve the fixed-length of C characters.

32. A method as in claim 29, wherein L is the number of characters in the alphabet.

33. A method as in claim 29, wherein the string is 48-bits long and the number portion is 32-bits long.

34. A method for offline-online handling of incentive points, comprising:

generating a code by:

providing a number portion, $S1_{INT}$, from a first string, S1

arranging a first secret key, K1, next to the number portion, $S1_{INT}$, from S1, to form a second string, S2,

applying a hash function to S2 to produce a third string, S3,

extracting a validation portion, $S1_{VAL}$, from S3 and arranging $S1_{VAL}$, next to $S1_{INT}$ in S1 ($S1 = S1_{VAL} + S1_{INT}$),

encrypting S1 using a second secret key, K2, to form a fourth string, S4, and

deriving the code by converting S4 to a base L fixed-length code string; and

fixing the code onto a medium such that the code is obtainable from the medium offline.

35. (New) A method as in claim 34, wherein the first and second secret keys, K1 and K2, are 128-bits long and the encryption involves DES3 encryption algorithm.

36. A method as in claim 34, wherein the hash function is MD5, a one-way hash algorithm.

37. (New) A method as in claim 34, wherein S1 is 48-bits long and the number portion, $S1_{INT}$, is 32-bits long.

38. (New) A method as in claim 30 wherein the step of verifying the submitted code includes,

converting the submitted code from a base L string into a base 2 string, $S4_{BASE2}$,

decrypting $S4_{BASE2}$ using a second secret key, K2, to form a decrypted first string, S1',

providing a number portion from S1'

arranging a first secret key, K1, next to the number portion from S1' to form a second string, S2',

applying a hash function to S2' to form a third string S3',

extracting a validation portion from S3' and a validation portion from S1', and

determining if the code is valid by comparing the validation portion from S3' with the validation portion from S1'.

39. (New) A method as in claim 38, wherein S3' and S1 are each 48-bits long and the secret keys, K1 and K2 are 128-bits long.

40. (New) A method as in claim 38, wherein the decryption involves $DES3^{-1}$ decryption algorithm and the has function involves MD5 hash algorithm.